



C1 - 2 变量与输入

// 学习目标

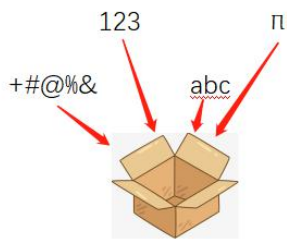
- 掌握变量的定义与使用
- 掌握 cin 输入语句
- 学会输出特定的小数位数

// 变量定义

变量是指在程序运行过程中，其值可以改变的量。

想想之前你在数学题里使用的未知数 x ，想求什么直接用 x 来表示它，也可以理解为我们把一个数字存放到了变量 x 里边。

所以你发现没？变量 x 是不是像一个存储数据的盒子？



要使用变量的话，我们要事先和计算机声明需要一个什么样的变量？这个动作也叫定义变量，因为计算机会把数据进行分类存储，大致有以下几种类型：

整型(int)	浮点型(float)	字符型(char)	布尔型(bool)
123、0、-123	3.141	'a' '+' '3'	true 或 false

// 如何定义变量呢？

一般我们先声明它的类型，然后再给变量起个名字，比如定义整数变量：

```
int a ;
```

这是我声明的一个存放整数的变量，变量名为 a ，同理我们还可以定义出其他的变量，比如：

```
float b ;
```

```
char c ;
```

```
bool d ;
```

如果需要很多整数变量，还可以进行连续定义。

```
int a , b , c ;
```

※注意：变量的名字**不能随便起名**哈！

变量名也称之为“**标识符**”，C++ 中的标识符命名规则如下：

- 标识符不能和系统里的关键字同名。比如 int、include 这些词汇！
- 标识符只能包括字母、数字和下划线“_”，并且开头不能是数字。
- 标识符必须先定义、后使用。
- 在同一作用域内，标识符不能重复定义。
- 标识符区分大小写，如 A 和 a、sum 和 Sum 都是不同的标识符。
- 标识符要尽量做到“见名知义”，增强程序的可读性。比如题中描述到


要求出多少钱的问题，那么我们可以定义一个变量名叫“money”，这样自己看或者别人看一下子就懂了！是不是很好呢？

// 给变量赋值

在 C++ 中，可以通过“赋值”语句来修改变量的值。使用“ = ” 作为赋值运算符，赋值语句的格式如下：

变量名 = 值或者表达式 ；

如：a = 1 ；

给变量赋值有两种方式，记好了 

第一种：int a = 1 ; // 把 1 存入变量 a 中，使用 a 就和使用 1 一样 ；

第二种：int a ;

cin >> a ; // 这是输入语句，等到程序运行之后自己输入才行，

输入什么，a 就是什么！

来看一道题了解一下如何应用吧！

问题：圆的半径是 3，求圆的面积！（ π 取 3.14159）

这个问题最简单的输出答案的方式是：

```
#include <iostream>

using namespace std ;

int main(){

    cout << 3.14159 * 3 * 3 ;

    return 0 ;

}
```

这段代码的功能非常局限，只能求出半径为 3 的圆的面积！所以我们可以使用变量来表示圆的半径，需要改变某些数值的时候，可以直接修改给变量赋的值！

可以这样写：

```
#include <iostream>
using namespace std ;
int main(){
    int r = 3 ;
    float pi = 3.14159 ;
    cout << pi * r * r ;
    return 0 ;
}
```

而如果题目变成这样：

问题：圆的半径是 r ，求圆的面积！（ π 取 3.14159）

输入：输入为一行一个整数 r ，表示圆的半径。

输出：圆的面积。

输入样例： 5

输出样例： 78.5397

这里提到了输入的格式要求，因此要用到 `cin` 输入语句，代码需要这样写：

```
#include <iostream>
using namespace std ;
int main(){
    int r ; // 这里定义出圆的半径就好，不需要赋值
    float pi = 3.14159 ;
    cin >> r ; // 这里是输入圆的半径的值，也是我们之前提到的第二种赋值方式
    cout << pi * r * r ;
    return 0 ;
}
```

// cin 输入语句

`cin` 是 C++ 中实现输入功能的语句，其格式一般为：

`cin >> 项目 1 >> 项目 2 >> ... >> 项目 n;`

在使用 `cin` 语句从键盘输入多个数据项的时候，一定要注意三个一致：

数据个数一致
输入顺序一致
数据类型一致

再看一道题，关于长方形面积计算

【问题描述】

一个长方形的长是 a 厘米,宽是 b 厘米, 请编程计算长方形的面积。

【输入格式】

一行两个整数 a 和 b , 表示长方形的长和宽。

【输出格式】

一行一个整数, 表示长方形的面积。

【输入样例】

4 2

【输出样例】

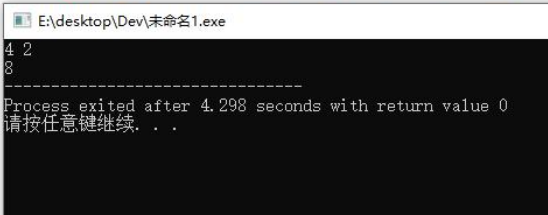
8

注意：这里的输入样例中有两个数据，分别代表长方形的长和宽，因此需要按照顺序逐个输入！

```
#include <iostream>
using namespace std ;
int main(){
    int a , b , s ; // 定义变量 a、b、s，分别表示长、宽、面积
    cin >> a >> b ; // 这里代表依次输入 a 和 b
    s = a * b ;
    cout << s ;
    return 0 ;
}
```

运行结果如下：

```
1 #include <iostream>
2 using namespace std ;
3 int main( ){
4     int a , b , s ; // 定义变量 a、b、
5     cin >> a >> b ; // 这里代表依次输
6     s = a * b ;
7     cout << s ;
8     return 0 ;
9 }
```



// 控制小数位数

之前我们有学习如何输出小数，即只要有小数参与运算即可，但普通的计算并不能按照自己的意愿来保留小数的位数，因此，我们可以选择保留几位小数！

使用方法如下：

```
#include <iostream>
#include <iomanip> // 需引入<iomanip>库，调用里边的设置精度功能
using namespace std ;
int main(){
    cout << fixed << setprecision (2) << 1.0 / 3 ;
    return 0 ;
}
```

如果有多个答案需要保留小数输出，可以这样写：

```
cout << fixed << setprecision (2);
cout << 1.0 / 3 << endl;
cout << 10.0 / 3 << endl;
cout << 100.0 / 3 << endl;
```

单词释义：

fixed (固定的)；

setprecision (设置精度)；